Each of the objects 230 preferably contains information about an individual data element of the cache 42 of Figure 1. In one embodiment, each object 230 stores a header 230 identifying the data element modeled by the object 230. Each object 230 may also store a history 234 of the represented data element. In one embodiment, the history 234 is represented by a priority value 236 assigned to the data element. A marker 238 indicating whether the data element 223 was stored as a result of a prefetch operation or not may also be stored within the object 230. A time stamp 240 indicating when a data element first entered the cache is also preferably present within each object 230. Of course, other data that may be needed to accurately model each data element that resides within the cache 42 of Figure 1 may likewise be stored in or with the objects 230.

IN THE CLAIMS

Please delete claims 5, 6, 9, and 10 without prejudice.

Please amend claims 1, 2, 7, 20, and 21, as follows:

1. A method for scheduling prefetches into a cache of a data storage system, the method comprising:

remotely modeling dynamic operation of the cache;

the remotely modeling including providing a model of data elements stored within the cache; and

-2-

A2

making a cache management decision based upon the model.

2. The method of claim 1, wherein making a cache management decision comprises:

intercepting a request for a data element from a stream of Input/Output (I/O) data requests passed between a host and a storage device of the data storage system; and

determining whether to schedule a prefetch of a data element logically successive to the requested data element in accordance with contents of the cache as indicated by the model.

A3

7. The method of claim 1, wherein remotely modeling the cache further comprises:

determining a size of the cache;

periodically fetching an I/O rate of the cache; and

periodically fetching a hit rate of the cache.

A4

20. A method for scheduling prefetches in a data storage system having a host and a cache, the method comprising the steps of:

providing a cache for caching Input/Output (I/O) data;

providing a prefetch module remote to the cache;

remotely modeling the cache within the prefetch module and determining whether to schedule a prefetch of data into the cache according to the results of the step of remotely modeling the cache, the step of remotely modeling the cache module further comprising:

examining the history of a data element in the cache;

assigning a priority value to the data element according to its history;

comparing that priority value to a predetermined threshold value;

determining a size of memory used in the cache;

periodically fetching an I/O rate of the cache from the cache;

periodically fetching a hit rate of the cache from the cache; and

determining a single reference residency time for a data element within the cache;

intercepting a stream of I/O information from the host to the cache to locate a requested data element;

determining if the requested data element in the stream of I/O information is already present within the cache;

making the requested data element a youngest member of the cache;

determining if the data element preceding the requested data element is present in the cache;

assigning a priority value to the requested data element;

periodically reevaluating the performance of the cache versus an internal model of the cache if the number of I/O requests received by the cache is greater than a predetermined number;

updating the dynamic threshold used in the internal model of the cache if the dynamic threshold value does not adequately model the performance of the cache;

-4-

comparing the priority value of the requested data element with the dynamic threshold value; and

prefetching the requested data element if the priority value of the requested data element is greater than the dynamic threshold value by passing an I/O request of the data element to the cache.

21. A data prefetch scheduling system comprising:

a cache configured to communicate with a host; and

a remote prefetch module configured to communicate with the host and the cache and configured to determine whether to schedule a prefetch of data into the cache; and

a modeling module operating within the remote prefetch module configured to model the cache.

Please add claims 29-32 as follows:

29. A prefetch module for determining whether to schedule a prefetch of data into a cache a computer system, the prefetch module comprising:

a modeling module configured to model dynamic operation of the cache;

wherein the modeling module is further configured to provide a model of data elements stored within the cache; and

a calculation module configured to make a cache management decision based upon the model.

30. A computer station on a computer network, wherein the computer station is configured to communicate with a cache coupled to a storage device of the computer network, the computer station comprising:

a processor; and

a memory configured to store data structures comprising:

a modeling module configured to model dynamic operation of the cache;

wherein the modeling module is further configured to provide a model of data elements stored within the cache; and

a calculation module configured to make a cache management decision based upon the model.

31. A computer readable medium comprising executable data structures configured to carry out a method for scheduling prefetches into a cache of a data storage system, the method comprising:

remotely modeling the dynamic operation of the cache;

the remotely modeling including providing a model of data elements stored within the cache; and

making a cache management decision based upon the model.

32. A data prefetch scheduling system comprising:

a means for caching data in communication with a host;

a means, in communication with the host and the means for caching data, for determining whether to schedule a prefetch of data into the means for caching data; and

a means, operating within the means for determining whether to schedule a prefetch of data into the means for caching data, for modeling the means for caching data.